USER BEHAVIOR MODELING IN PEER-TO-PEER FILE SHARING NETWORKS: DISSECTING DOWNLOAD AND REMOVAL ACTIONS

Qinyuan Feng[*], *Yu Wu*[*], *Yan Sun*[†], *Jing Jiang*[*], *Yafei Dai*[*]

[*]CNDS Lab, Peking University, Beijing, China
{fqy,wuyu,jiangjing,dyf}@net.pku.edu.cn
[†]University of Rhode Island, Kingston, RI, USA
yansun@ele.uri.edu

ABSTRACT

User behavior models are important for building realistic simulation environment for research on P2P multimedia file-sharing systems. In this paper, we build a user download behavior model and a user removal behavior model, which can describe important user characteristics that are not captured in the existing models. The proposed download behavior model incorporates retry behavior, and the removal behavior model integrates free-riding, file usage, and file removal. Based on two-year real user logs, we derive the range of all the model parameters and generate many interesting observations. To validate the proposed models, we compare several models in a case study on the number of living replicas in P2P file-sharing systems. The results demonstrate the accuracy, usability, and advantage of the proposed models.

Index Terms— Modeling, Multimedia systems

1. INTRODUCTION

For the research on P2P multimedia file-sharing systems, new algorithms and schemes are mainly evaluated through *simulations* because experimental testing in real P2P networks is prohibitively expensive [1]. Therefore, building realistic simulation environment is critical.

The major obstacle toward building the realistic simulation environment is the lack of a throughout understanding of dynamic behaviors of P2P users, which are affected by technology as well as human factors. The most important behaviors are file download and file removal. Most of the existing user behavior models, however, cannot capture two important factors in file download and file removal behaviors.

- **retry behavior:** how users retry after download failure;
- **retention time:** the time interval between a user downloading a file and removing it out of the P2P file-sharing system. It is affected by (i) free-riding, (ii) delay between downloading and examination of the files, and (iii) additional retention time after examination.

The above two factors can greatly affect simulation results. However, it is challenging to understand and model these factors because it is hard to capture retry behavior and retention time directly from most of the current P2P systems. To our best knowledge, none of the existing work have parameterized these two factors from real user logs.

In this paper, we build user behavior models based on two-year user logs in a real-world P2P network that is mainly used to exchange video files. The models include statistical values of all parameters. This work can be used directly to generate realistic user behaviors in simulations for P2P research. More important, the proposed models reveals interesting user behaviors and can assist research on improving efficiency and usability [2], on designing trust and incentive mechanisms, and on identifying fake files in P2P networks [3]. We also conducted a case study on calculating the number of living replicas in P2P systems. Whereas other models significantly overestimate the number of living replicates, the proposed models yield much more accurate results.

The rest of the paper is organized as follows. Section 2 discusses the related work and Section 3 discusses the methodology. The download and removal behavior models are presented in Section 4 and Section 5, respectively. The case study is shown in Section 6, followed by the conclusion in Section 7.

2. RELATED WORK

There are many existing works about network and user behaviors modeling in P2P systems. For instance, Schlosser et al. [1] proposed a query-cycle simulator concentrating on traffic and network behaviors. Ge et al. [4] proposed a simple model to describe the effects of scaling, freeloaders, file popularity and availability. Tutschku et al. [5] concentrated on the traffic characteristics and performance evaluation of P2P systems. Lee et al. [6] found a bimodal distribution of the time interval between download and quality checking. Handurukande et al. [7] presented an empirical study on a workload gathered by crawling the eDonkey network, confirmed the prevalence of free-riding and the Zipf like distribution of file popularity, and analyzed the evolution of document popularity. Tian and Dai [8] produced a thorough measurement of the dynamic nature of P2P systems. In addition, there are also some work [9, 10] concentrating on the BitTorrent-type systems. However, the gap between the existing user behavior models and the reality still exists. Some of the existing models are too simple to capture many critical behaviors of real users, such as retry, whereas many other models make assumptions on the values of critical parameters without empirical studies in real networks.

3. METHODOLOGY

The models developed in this paper are based on real user logs in MAZE [8], a popular P2P file-sharing system with more than 2 million registered users and 40,000 users online at peak times. The dominating files in MAZE are multimedia data. Our previous work [8, 3] has demonstrated that the user behaviors in MAZE are similar to those in many other P2P file-sharing systems.

We have analyzed hundreds of files that were randomly selected from the logs. Due to the space limitation, we only illustrate the results for five representative movie files, and use F_i for $i = 0, 1, 2, 3, 4$ to represent them. In this study, we first develop

THIS WORK IS SUPPORTED BY NGFR 973 PROGRAM (NO. 2004CB318204), NSFC (NO. 60673183), DOCTORAL FUNDING OF MOE (NO. 20060001044).

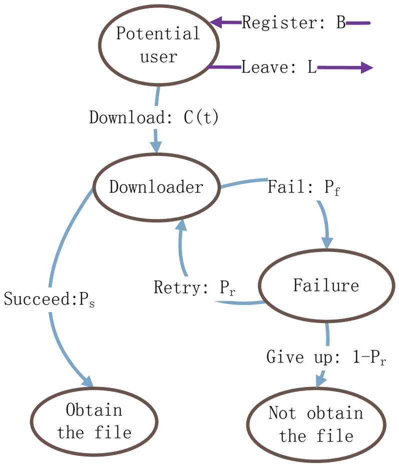

Fig. 1. State change of download model

models containing multiple states, then obtain all the key parameters from real user logs, and finally integrate all the models together in a case study.

4. DOWNLOAD BEHAVIOR MODELING

In this section, we propose a download behavior model that incorporates the *retry behavior* in case of download failure. Figure 1 illustrates this model for individual files.

The ***potential users*** of a file refer to the users in the system that have never tried to download this particular file. When a file is first published, all the users in the system are potential users. Note that most of the potential users may never try to download this file. As time goes by, the number of potential users can increase due to new user registration (with rate B) and decrease due to user departure (with rate L) or download (with rate $C(t)$). When a potential user tries to download a file, he becomes a ***downloader*** of this file. A downloader may try many times to download a file. An attempt succeeds with probability P_s and fails with probability $P_f = (1 - P_s)$. After a download attempt fails, a user will attempt to download the file again (retry) with probability P_r, and will give up totally with probability $(1 - P_r)$.

Identifying the basic elements in this model is not difficult. The challenging task is to determine the model parameters, including: P_s, P_r, $C(t)$, and $U(t)$, where $U(t)$ is the number of potential users at time t and $C(t)$ is the download probability.

4.1. Success, failure and retry rates

Most of the existing models only capture the probability that a user successfully downloads a file eventually, without getting into the details of user retrying and giving up behaviors. We concentrate on the success and failure of each download attempt.

The notations in this subsection are as follows.

- u_i: the number of users who have failed to download the file in the previous $(i-1)$ attempts and will try to download the file for the i^{th} time. u_1 is just the total number of downloaders who ever try to download the file.
- us_i: the number of users who successfully download the file at the i^{th} attempt.
- us: the number of the users who successfully download the file eventually, and $us = \sum_{i=1}^{\infty} us_i$.
- P_{us}: the percentage of users who successfully download the file eventually and $P_{us} = \frac{us}{u_1}$.

Based on the definition of u_i, us_i, and us, we obtain

$$u_{i+1} = u_i \cdot P_f \cdot P_r \text{ and } us_i = u_i \cdot P_s \quad (1)$$

From (1), we derive

$$P_r = \frac{1 - P_s/P_{us}}{1 - P_s} \quad . \quad (2)$$

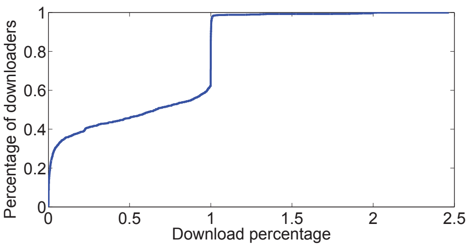

Fig. 2. CDF of the cumulated downloaded percentage, for file F_0

Table 1. Download failure and retry behavior parameters

	u_1	P_{us}	P_s	P_f	P_r
F_0	1525	38.0%	27.1%	72.9%	39.3%
F_1	1326	39.4%	25.6%	74.4%	47.1%
F_2	1104	40.0%	26.4%	73.6%	46.1%
F_3	9945	66.5%	59.0%	41.0%	27.6%
F_4	21211	63.7%	58.2%	41.8%	20.4%

Due to the limitation of the user logs, we cannot obtain the critical model parameters P_r and P_s directly. Instead, we can obtain the *cumulated downloaded percentage*(CDP) from the user logs. Here, CDP of one user for a specific file is defined as the total downloaded size of this file divided by the size of this file. For example, a file is 100MB. A user downloaded 70MB at the first attempt. The first attempt failed. Then, the user downloaded 100 MB at the second attempt. So, this user's CDP for this file is 1.7.

Figure 2 shows the cumulative density function (CDF) of CDP for F_0. The x-axis means CDP and y-axis means the CDF of CDP. From this figure, we observe that

- about 60% of the downloaders' CDP values are less than 1, which indicates that they do not get the complete file.
- about 2% of the downloaders' CDP values are larger than 1, which indicates that they have downloaded more than once.
- for the users whose CDP value is 1, it is highly likely that they successfully download this file at the first attempt.

We derive the critical model parameters from the cumulative density function of CDP as follows. First, by setting $i = 1$ in (1), we get $P_s = us_1/u_1$. That is, P_s is just the success rate of the first attempt. Therefore, we can estimate P_s as the percentage of downloaders whose CDP value is 1. Second, we approximate P_{us} as the percentage of downloaders whose CDP value is greater than or equal to 1. It is noted that this approximation overestimates P_{us} because with a small probability, a user with CDP ≥ 1 may not have a successful download. Finally, with P_s and P_{us}, we calculate P_r using (2). The critical model parameters for 5 representative files are shown in Table 1.

It is seen that the success rate (P_s) in MAZE is low. However, there are few users complaining about it. This shows that the users of MAZE, and possibly other P2P file-sharing systems, have a high level of tolerance of failures. Furthermore, *a large percentage of users, ranging from 20.4% to 47.1%, will retry after experiencing download failures. Retry is a very important feature of download behaviors, but it is not well captured in the existing models.* Our work provides a way to quantify this feature.

4.2. Potential User Set

We can obtain the number of new downloaders each day of a file as a function of time from user logs. This value is represented by $D(t)$. Recall that $U(t)$ denotes the number of potential users. We have

$$U(t + \Delta) - U(t) = B(t) - U(t) \cdot L - D(t). \quad (3)$$

Here, $B(t)$ is the number of new users registered between time t and $t + \Delta$, and Δ is chosen as one day in our work. L is the probability

Table 2. Parameters of download model

	B	$C(t_0)$	α	$U(t_0)$	β
F_0	1665	0.107%	13.2%	245,509	2
F_1	1665	0.092%	9.9%	245,509	1
F_2	1665	0.079%	5.8%	245,509	1
F_3	1174	0.126%	1.1%	253,459	3
F_4	1237	0.178%	0.1%	265,774	3

that a user will leave the system permanently, so the term $U(t) \cdot L$ approximates the number of users who leave the system permanently. D(t) represents the number of potential users who begin to download the file between time t and $t + \Delta$.

From (3), we see that if a file is first published at time t_0, $U(t)$ will be uniquely determined by four factors: $B(t)$, L, $D(t)$, and $U(t_0)$ that is the initial size of the potential user set. Particularly, the value of $U(t_0)$ can be estimated directly from user logs as the number of users who are still in the system, including the users who are offline and will be online later, at t_0. The values of $B(t)$ are directly obtainable from user logs. Additionally, in our previous study [8], we have found that $L \approx 3.6\%$ per day in MAZE. More details will be presented in the following section.

4.3. Download Rate

In this section, we determine $C(t)$, the fraction of the potential users that turn into downloaders at time t.

In the proposed model, we obviously have

$$C(t) = D(t)/U(t) \tag{4}$$

Recall that $U(t)$ can be calculated from $U(t_0)$, $B(t)$, and $D(t)$. Then, we calculate $C(t)$ in three steps.

1. From user logs, find all the users that registered before t_0 and would be online at least once after t_0. $U(t_0)$ is estimated as the number of such users.
2. Estimate B(t) as the average number of newly registered users per day, which can be calculated directly from the user logs.
3. Solve $U(t)$ using (3) and then solve $C(t)$ using (4).

After examining the $C(t)$ values for many files, we observe that $C(t)$ reaches its maximum value on day $t_0 + \beta$, where β is smaller than 3 days for most of the files. For F_0, $\beta = 2$. Additionally, $C(t)$ decreases to a small but stable value after a certain amount of time.

This phenomenon is easy to understand because users are more likely to download a file when it is first published and become less interested as time goes. In fact, from the definition of β, we know the maximum value of $C(t)$ is

$$C(t_0 + \beta) = D(t_0 + \beta)/U(t_0 + \beta). \tag{5}$$

Then, we use a nonlinear decreasing function to model $C(t)$ as

$$\frac{d(C(t))}{dt} = (c_{lb} - C(t)) \cdot \alpha\,, \tag{6}$$

where c_{lb} $(= C(\infty))$ is a very small value close to 0. In this work, we set $c_{lb} = 1e{-}5$. And, α represents the speed of convergence and is the only parameter to be determined now. We set the time interval as one day and transform (6) into

$$C(t + x) = (1 - \alpha)^x (C(t) - c_{lb}) + c_{lb}\,. \tag{7}$$

By solving (7), we can get

$$\alpha = 1 - \sqrt[x]{\frac{C(t + x) - c_{lb}}{C(t) - c_{lb}}}\,, \tag{8}$$

Then, we solve α from (8) by setting $t = t_0 + \beta, x = 30$, and use the $C(t)$ and $C(t + x)$ values calculated from the user log files as described earlier.

Table 2 shows the parameters for the five movie files. The first three files were popular movies published on the same day, so their B and U_{t_0} are the same. The last two files are adult videos, whose popularity can be stable for a long time, so their α is very low. *It is seen that nearly 0.107% of the potential users can be interested in downloading F_0 (a popular movie) on the first day of release.*

To validate the download behavior model, we reconstruct $D(t)$ based on the parameters in Table 2, and compare it with the directly measured data. Good match is observed. (Figures are not shown due to space limitation.)

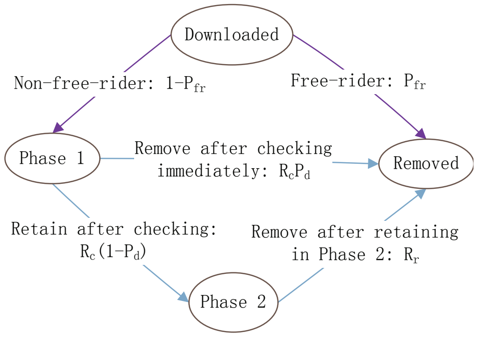
Downloaded
Non-free-rider: 1-P$_{fr}$
Free-rider: P$_{fr}$
Phase 1
Remove after checking immediately: R$_c$P$_d$
Removed
Retain after checking: R$_c$(1-P$_d$)
Remove after retaining in Phase 2: R$_r$
Phase 2

Fig. 3. State change in removal behavior model

5. REMOVAL BEHAVIOR MODELING

Quantitative models for user removal behavior are extremely rare in the current literature. In this section, we present a user removal behavior model with realistic model parameters, and then derive the ***retention time***, which is the time interval between a user downloading the file and removing it out of the system. Here, removing means that the file is moved out of the "shared" folder and cannot be seen by other P2P users.

As shown in Figure 3, the proposed model has four states: *download*, *removed*, *store before checking* (referred to as Phase 1), and *store after checking* (referred to as Phase 2). In Phase 1, the file has been downloaded but the user has not checked or used the file yet. In Phase 2, the user has checked or used the file but has not removed the file yet. The state change is described as follows.

- After downloading a file, a user will move it out of the file-sharing system with probability P_{fr}. This is a free-riding behavior, and the retention time of free riders is 0. Then, with probability $1 - P_{fr}$, the user moves to Phase 1.
- The user in Phase 1 will use and check the file with rate of R_c. After use/check, a user will delete the file with probability P_d and will continue to store it with probability $1 - P_d$. Thus, the transition probability from Phase 1 to the remove state is $R_c P_d$, and from Phase 1 to Phase 2 is $R_c(1 - P_d)$.
- In Phase 2, a user will delete the file with the probability of R_r.

For a user who has downloaded a specific file, let $PH_1(t)$ be the probability that he/she is in Phase 1 at time t, and let $PH_2(t)$ be the probability that he/she is in Phase 2 at time t. In addition, we use $P_{rt}(t)$ to represent the probability that a user's retention time is equal to or smaller than t (i.e. CDF of the retention time). It is straightforward that

$$PH_1(0) = 1 - P_{fr}\ ,\ \ PH_2(0) = 0\ ,\ \ P_{rt}(0) = P_{fr}\ . \tag{9}$$

Since a user will leave Phase 1 with rate R_c, we get

$$\frac{d(PH_1(t))}{dt} = -PH_1(t) \cdot R_c\ . \tag{10}$$

For Phase 2, the probability of moving out is $PH_2(t) \cdot R_r$ and the probability of moving in is $PH_1(t) \cdot R_c \cdot (1 - P_d)$. We have

$$\frac{d(PH_2(t))}{dt} = PH_1(t) \cdot R_c \cdot (1 - P_d) - PH_2(t) \cdot R_r\ . \tag{11}$$

Similarly, we have

Table 3. Parameters in the Removal Behavior Model

	P_{fr}	R_c	P_d	R_r
F_0	50.4%	93.6%	77.8%	16.1%
F_1	60.2%	82.2%	90.7%	15.0%
F_2	56.8%	93.4%	73.9%	6.0%
F_3	57.0%	83.3%	92.9%	12.6%
F_4	48.6%	95.5%	81.6%	5.6%

$$\frac{d(P_{rt}(t))}{dt} = PH_1(t) \cdot R_c \cdot P_d + PH_2(t) \cdot R_r \ . \qquad (12)$$

From (10) - (12), $P_{rt}(t)$ can be solved as long as we know P_{fr}, R_c, R_r, and $R_c \cdot P_d$. Next, we discuss how to obtain these parameters from user logs.

- Let N_d denote the number of users who have ever downloaded the complete file. Among these users, from the logs, we can identify the free-riders who have never uploaded a file. The number of free riders is N_{fr}. Then, we get $P_{fr} = N_{fr}/N_d$.
- The results in [6] as well as our analysis show that most users check the downloaded files during the first day after downloading. Thus, if a file is removed during the first day after downloading, it is highly likely that the file is removed from Phase 1; if a file is removed during the 2^{nd} and 3^{rd} day, this file can be removed either from Phase 1 or Phase 2; and if a file is removed after the 3^{rd} day, it is highly likely that this file is removed from Phase 2. Therefore, we estimate $R_c \cdot P_d$ as the removal rate during the first day, and R_r as the average removal rate between the 4^{th} day and the 15^{th} day.
- Then, we estimate $P_{rt}(1)$ as the number of users who remove the file within the first day divided by N_d.
- Finally, from (10) - (12), we can derive

$$R_c = \frac{(R_c \cdot P_d) \cdot (1 - R_r) + P_{rt}(0) - P_{rt}(1)}{(R_c \cdot P_d) - R_r}. \qquad (13)$$

We have already estimated all variables at the right hand side of (13). Thus, R_c is determined by (13).

The estimated model parameters are shown in Table 3. *We can see that (1) about half of the users are free-riders; (2) about 90% of the users will check the downloaded file within one day; (3) about 80% of the users will delete the files after using them once, and (4) the others will delete the files with the rate of around 10% per day.*

We have compared the reconstructed cumulative density function of the retention time with directly measured data, good match is observed. (Figures are not shown due to space limitation.)

6. CASE STUDY

The proposed models can capture the details of user behaviors. They can be used in many applications. In this section, we provide a case study to illustrate the usage and the advantage of the proposed models.

In P2P systems, estimating the number of living replicas is important, such as in the cache design [2]. A replica is alive if (1) it has been downloaded successfully, (2) it has not been removed, and (3) the downloader has not left the system permanently. Using the proposed models, we calculate the number of living replicas as

$$Rep(t) = \int_{t_0}^{t} D(x)P_{us}[1 - P_{rt}(t - x)]e^{-L\cdot(t-x)}dx. \qquad (14)$$

Here, $D(x)P_{us}$ is the number of successful downloads at time x, $1 - P_{rt}(t - x)$ represents the probability that a downloader downloads the file at time x and will still retain it at time t, and $e^{-L\cdot(t-x)}$ represents the probability that a user is in the system at time x and is still in the system at time t.

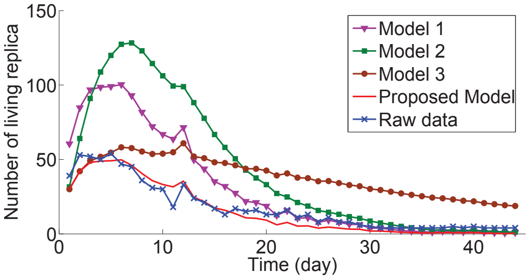

Fig. 4. Number of living replicas changing with time

The estimated results and the directly measured results are shown in Figure 4. Most of the existing work does not consider all of these factors, so we also compare the proposed model with the model without considering free-riding (Model 1, $P_{fr} = 0$) [1], the model without considering the immediate removal after checking (Model 2, $P_d = 0$) [7], and the model without considering the removing in Phase 2 (Model 3, $R_r = 0$) [6]. It is seen that the proposed model fits the raw data very well, but all other three models overestimated the number of living replicas, due to the underestimation of users' removal behaviors. We can see that all the components in the proposed models are important. Missing any of them will lead to large bias in the estimation results.

7. CONCLUSION

In this paper, we build user behavior models that have two major advantages. First, they describe important characteristics that are not captured in any single model in the current literature. In particular, our download model considers retry behavior and our removal behavior model considers the probability of free-riding, file checking, and file removal. Second, all the model parameters are derived from real user logs, such that the models are ready to be used. In the case study, compared with other models, the proposed models generate much better estimation of the number of living replicas in P2P systems. All components in the proposed models play important roles in generating the accurate estimation.

8. REFERENCES

[1] M. T. Schlosser, T. E. Condie, and S. D. Kamvar, "Simulating a File-Sharing P2P Network," in *Proc. of SemPGrid*, Budapest, May 2003.

[2] A. Wierzbicki, N. Leibowitz, M. Ripeanu, and R. Wozniak, "Cache replacement policies revisited: The case of P2P traffic," in *Proc. of the CCGrid*, Chicago, April 2004.

[3] Q. Feng and Y. Dai, "LIP:A Lifetime and Popularity Based Ranking Approach to Filter out Fake Files in P2P File Sharing Systems," in *Proc. of IPTPS*, Bellevue, Febrary 2007.

[4] Z. Ge, D. R. Figueiredo, S. Jaiswal, J. Kurose, and D. Towsley, "Modeling peer-peer file sharing systems," in *Proc. of INFOCOM*, San Francisco, April 2003.

[5] K. Tutschku and P. Tran-Gia, "Traffic Characteristics and Performance Evaluation of Peer-to-Peer Systems," *Peer-to-Peer Systems and Applications*, LNCS, vol. 3485, pp 383-397, 2005.

[6] U. Lee, M. Choi, J. Cho, M. Y. Sanadidi, and M. Gerla, "Understanding Pollution Dynamics in P2P File Sharing," in *Proc. of IPTPS*, Santa Barbara, February 2006.

[7] S. B. Handurukande, A. Kermarrec, F. L. Fessant, L. Massoulie, and S. Patarin, "Peer sharing behaviour in the eDonkey network, and implications for the design of server-less file sharing systems," in *Proc. of EuroSys*, Leuven, April 2006.

[8] J. Tian and Y. Dai, "Understanding the Dynamic of Peer-to-Peer Systems," in *Proc. of IPTPS*, Bellevue, Febrary 2007.

[9] D. Qiu and R. Srikant, "Modeling and performance analysis of bit torrent-like peer-to-peer networks," in *Proc. of SIGCOMM*, Portland, August 2004.

[10] L. Guo. S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang, "Measurements, Analysis, and Modeling of BitTorrent-like Systems", in *Proc. of IMC*, Berkeley, October 2005.